1.   A method of formatting a plurality of strings of data derived from respective ones of a plurality of data processors, comprising:

5   initializing the data in each of said strings;

finding the differences between said strings; and

providing said differences in a display to a user.

2.   The method of Claim 1, wherein the act of initializing comprises:

creating a symbol table having a plurality of symbol numbers associated

10   therewith;

creating said symbol array having at least one element for each of said strings;

for each of said input strings, determining whether said each string is present in said symbol table; and

15   if said each string is present in said symbol table, obtaining at least one symbol number for said string from said symbol table.

3.   A method of analyzing a plurality of strings of data derived from at least one data processing device, comprising:

initializing said data, said act of initializing including creating a symbol

20   array;

analyzing said strings of data based at least in part on said symbol array; and

identifying at least one relationship between one or more of said data within one or more of said strings.

25   4.   The method of Claim 3, wherein the act of initializing comprises:

creating a symbol table having a plurality of symbol numbers associated therewith;

creating said symbol array having at least one element for each of said strings;

30   for each of said input strings, determining whether said each string is present in said symbol table; and

if said each string is present in said symbol table, obtaining at least one symbol number for said string from said symbol table.

5.　A method of identifying differences within a plurality of data inputs associated with at least one data processor, comprising:

providing a plurality of data inputs having at least one data string associated therewith;

processing those inputs of said plurality that do not share any of said at least one data strings with other of said inputs;

determining whether an excluded set of data is empty; and

if said excluded set of data is not empty, identifying said differences using respective members of an included set as said plurality of inputs.

6.　The method of Claim 5, wherein when said excluded set is empty, the method further comprising:

identifying groups of contiguous strings of data which are identical in all of said inputs;.

determining the order of said groups of contiguous strings of data; and

determining whether said groups of contiguous strings are present in the same order in all of said inputs.

7.　The method of Claim 6, further comprising:

identifying those inputs having strings preceding the first of said groups; and

identifying differences among that subset of said inputs which consists of said strings which precede the first group in each input.

8.　The method of Claim 7, further comprising determining the differences among that subset of said inputs that consists of the strings between a first group and a second group.

9.　The method of Claim 5, wherein the act of processing comprises:

clearing a symbol counts array;

for each of said strings in each of said inputs, incrementing the symbol count;

for each of said string in each of said inputs, determining the symbol count for that string;

and for each of said inputs, adding the input to an excluded set.

10.   The method of Claim 6, wherein the act of identifying groups of contiguous strings of data identical in all of said inputs comprises:

creating a list of anchors containing strings that occur at a predetermined

5   frequency;

for each input of each of said anchors, generating a new group and associating each anchor string with said new group;

for each input of each of said anchors, associating each string which precedes said new group and which is identical in all inputs with said new group;

10   and

for each input of each of said anchors, associating each string which follows said new group and which is identical in all inputs with said new group.

11.   A storage device, comprising:

a computer readable medium;

15   a computer program stored on said computer readable medium, said program being adapted for analyzing a plurality of strings of data derived from at least one data processing device according to the method comprising:

initializing said data, said act of initializing including creating a symbol array;

20   analyzing said strings of data based at least in part on said symbol array; and

identifying at least one relationship between one or more of said data within one or more of said strings.

12.   A data processing device, comprising:

25   a processor adapted to process digital data and execute a computer program;

a storage device in data communication with said processor, said storage device comprising:

a computer readable medium; and

a computer program stored on said computer readable medium, said program being adapted for analyzing a plurality of strings of data derived from at least one data processing device according to the method comprising:

initializing said data, said act of initializing including creating a symbol array;

analyzing said strings of data based at least in part on said symbol array; and

identifying at least one relationship between one or more of said data within one or more of said strings.

13. The device of Claim 12, further comprising a plurality of data interfaces adapted to receive said data strings from respective ones of a plurality of software processes running on respective ones of a plurality of data processors.

14. The device of Claim 13, further comprising said plurality of data processors.

15. A method of designing a processing device adapted to run at least one software process thereon, comprising:

generating a design for said processing device;

running at least a portion of said at least one software process a first time;

obtaining a first output from said at least one process;

modifying said design;

running said at least portion of said at least one software process a second time;

obtaining a second output from said at least one process; and

identifying differences within said first and second output.

16. The method of Claim 15, further comprising comparing said differences to at least one predetermined criterion; and

evaluating the acceptability of said modifications based at least in part on said act of comparing.

17. A method of evaluating the operation of a plurality of software processes running on respective ones of a plurality of digital processors, comprising:

generating a first data string using a first of said plurality of software processes;

generating a second data string using a second of said plurality of software processes;

inputting said first and second data strings into a debug software process;

analyzing said first and second data strings using said debug process; and

evaluating the operation of said processes based at least in part on said act of analyzing.

18.   The method of Claim 17, wherein the act of analyzing comprises:

(i)   identifying common patterns within at least one first portion of said first and second data strings; and

(ii)   identifying differences within at least one second portion of said first and second data strings.

19.   A multi-processor integrated circuit device, comprising:

a first processor core adapted to run a first software process, said first software process adapted to generate a first string of data;

a second processor core adapted to run a second software process said second software process adapted to generate a second string of data;

at least one data interface, wherein said first and second processors respectively transfer data comprising said first and second strings to an external debug process adapted to identify similarities and differences between the operation of said first and second softare processes on said first and second processors, via said at least one interface.